# Natural Language processing based Automatic Text Summarization

Buragadda Manikanta Lokesh (Author)
Computer Engineering (Data Sciences)
Presidency University Bangalore

Anirudh K N (Author)
Computer Engineering (Data sciences)
Presidency university Bangalore

Dokku Sri Sai Abhiram (Author)
Computer engineering (Data Sciences)
Presidency University Bangalore

N. Nasurudeen Ahamed (Author)
Assistant Professor, CSE,
Presidency University, Bangalore

Abstract

*Text summary is a well-known technique for distilling a document's primary points. Beginning with data extraction from a website link, the recommended approach for text summarizing and keyword extraction continues through a number of steps. It will help with the creation of a machine learning and natural language processing solution to replace the existing product evaluation process (NLP). The next major one is Automatic Text Summarization (ATS), which may simply summarize the source data and provide us with a condensed form that preserves the content and general meaning. Automatic Text Summarization was proposed in the 1950s, but the discipline is still in its infancy. Undirected graphs, weighted graphs, keyword extraction, and sentence extraction are all used in the Text Rank algorithm. In this paper, we will look upon. The proposed solutions for text summarization and keyword extraction passes through a series of processes, beginning with information or data extraction through the website link, eliminating outliers and improper information, and constructing a summary of the extracted information or data. It will assist in the development of a machine learning and natural language processing solution to the traditional product review procedure (NLP).*

Key Words: *Text Summarization, relevant Information, summary, Natural language processing (NLP), Text Summarization.*

## *Introduction*

As more people join, the complexity grows. More semantic linkages are produced because of text Summarizers can be characterized as generic, specific. when the model is free of bias and past knowledge when it comes to the input domain-specific knowledge, in which the model makes use of domain-specific knowledge. to provide a more accurate summary depending on the information facts are well-known Query-based, with the summary being the only thing displayed provides well-known responses to inquiries posed in natural language the text that is being entered Summarizers are classed as follows based on output type: Extractive, in which key sentences are culled from the rest of the document. To make a summary, enter text. Where the model is abstract, to provide a more complete picture, it creates its own words and sentences. a logical summary, as if it were generated by a person In Abstract summaries are a more difficult endeavor in general.

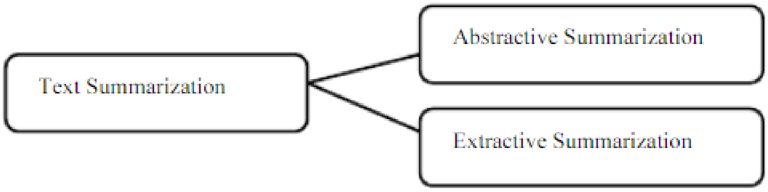

*Fig: Types of Text Summarization*

## Related Work

Text summary is a well-known technique for distilling a document’s primary points. The purpose of this research is to investigate extractive and abstract approaches for summarizing texts. The proposed system is largely focused on scraping data from websites and producing a summary as well as keywords based on that. The nest major is Automatic Text Summarization (ATS). Which may simply summarize the source data provide in with a condensed form that preserves the content and general meaning. Extractive and Abstractive Text Summarization are the two approaches used by ATS.

a well-known technique for condensing the main ideas of a publication. The goal of this study is to look into extractive and abstract techniques to text summarization. The suggested system is primarily concerned with scraping data from websites and generating a summary as well as keywords from that data. The suggested technique begins with pre-processing and lemmatization. Then, for text representation, a graph is constructed using nouns as nodes and non-noun terms as edges.

The assumption for picking sentences is that all nouns reflect distinct themes. The following big one is Automatic Text Summarization (ATS), which may simply summarize the source material and offer us with a condensed version that retains the content and basic meaning.

➢ Extractive:

Extractive Supervised summarization techniques reduce the weight of summarizing by selecting subsets of sentences. Extractive processes are designed as binary classification problems with the goal of identifying article phrases that belong in the summary. Supervised learning techniques need a large amount of identified information or labelled datasets. Extractive processes choose the top N sentences that best address the article's essential themes. NLP analysts are particularly interested in extractive summarization. single document and graph-based extractive system called EdgeSumm.

A set of such features are used for each phrase to provide the context locally and globally, which is described below.

1) AbstractROGUE: It is used for summarization as a feature. It uses the abstract, a pre-existing description, to manipulate the known structure of a paper. AbstractROUGE 's theory is that sentences that are strong qualitative summaries are often likely to have good summaries of the highlights.

2) Numeric Count: This is basically calculated on counting the times of numeric occurrences in a sentence, as sentences containing numbers/math do not contribute to a healthy summary.

3) Title Score: Nonstop words in the text which match with those in the title are given more importance in the summary.

4) Key phrase: Score Keywords used or predefined by the author are given more importance in the summary when used in the text.

5) Sentence Length: We add as an attribute the length of the phrase, an effort to catch the intuition that short phrases are quite unlikely to be successful summaries because they do not communicate as much data as longer phrases

- Abstractive:

Since sentences that form a group in a vector space may be close to each other, it is sufficient to keep one representative in each such group to make a summary Schumann [5] introduces an uncontrolled method of summarizing sentences in a logical way using the Variational Autoencoder (VAE). VAEsare trained in learning to reconstruct the input from potential variables. Instructing the decoder to produce a short output sequence leads to the output of the input sentence in a few words. TheVAE system uses text data using RNNs such as encoder and decoder. The idea that upgrading the decoder to produce shorter results will lead to more details expressed in a few words can be confirmed in the summary test. Linear Regression tests have shown that the length of the input sentence The model does not provide a viable solution to a large problem (as there are only a few rules for retrenchment), but it does recommend that group sentences presented at the top of the vector find groups of similar phrases and choose representatives for these groups to construct a summary.

The decoder is programmed to define sentence embedding. They utilize a mix of Sentence Embedding using RNN via Long Short-Term Memory (LSTM) and Sentence Embedding using RNN via Long Short-Term Memory (LSTM), in which they use a repeating neural network with short-term memory to identify phrase embedding.

The purpose is generally to embed sentences directly or indirectly in such a way that the sentences closest to the definition are embedded adjacent to each other in the vector space when expressing sentences at a high vector level.

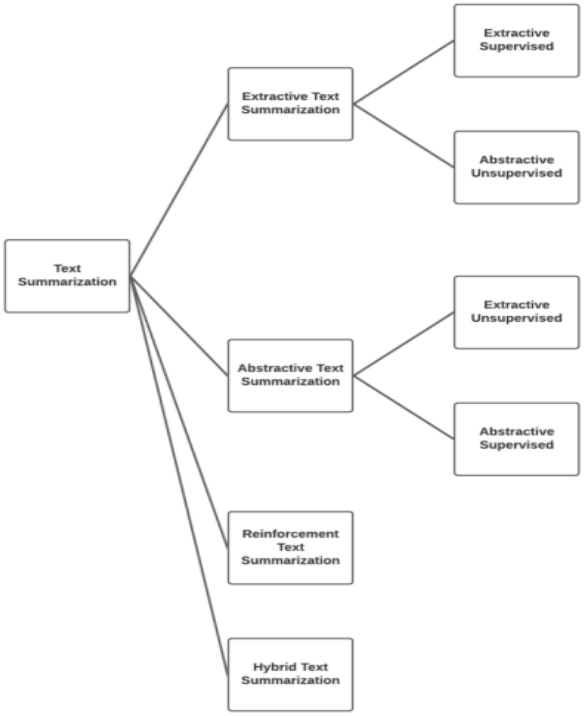
Text Summarization
Extractive Text Summarization
Extractive Supervised
Abstractive Unsupervised
Abstractive Text Summarization
Extractive Unsupervised
Abstractive Supervised
Reinforcement Text Summarization
Hybrid Text Summarization

*Fig: Taxonomy of Text Summarization*

We have covered a variety of supervised abstractive approaches in this study. The long-term dependencies are not captured by the basic RNN. The Long Short-Term Memory (LSTM) RNN model is used to correct this. An input gate, output gate, and forget gate make up an LSTM. Implements the Abstractive Contextual RNN (AC-RNN), which takes a document context vector as input to the encoder in the first phase. As a sequence labelling job, a Supervised Abstractive Model with Conditional Random Fields (CRF) compression is used. For sequence labelling, it employs the BIO labelling scheme.

Finally, this study considers many approaches to dealing with text outline, such as statistical methods, lexical chain-based methodologies, cluster-based methodologies, and fuzzy rationale-based methodologies.

**Methodologies:**

➢ Text Rank Algorithm

The text rank algorithm is a diagram-based positioning model for text processing that may be used to locate the most relevant phrases in a text as well as to find keywords. The process for determining text rank is comparable to the algorithm for determining page rank. In web search findings and web use mining, the page rank algorithm is used to designate Webpages. Sentences are taken into account in the text rank algorithm for ranking Web sites.

1. Identify the content units that best describe the present work and add them to the diagram as vertices.

2. Identify the relationships that connect the content units and use these relationships to build edges between vertices in the chart. Unweighted or weighted edges might be undirected or coordinated.

3. The diagram-based placement procedure is then looped until union is reached.

4. Mastermind the vertices based on their previous score. Choices for placement and determination Use the attributes that have been attached to each vertex. 5. Finally, a summary will be formed from the highest-level sentences.

➢ Text Rank Model:

Graph-based algorithms are still the most usual method for estimating the strength of a vertex in a graph, based on all of the information obtained from the whole graph. The vote and recommendation concept are at the heart of what we've done here. The score is connected to the vertex based on the votes cast. We use the "random surfer model" to calculate the likelihood of jumping from one vertex to another. The score of a graph is computed iteratively, starting with arbitrary numbers. The significance of a vertex is used to determine its score, while the latter traits are unaffected by the first.

➢ Centroid based Summarization

Another set of approaches that has become a popular baseline is centroid-based summarization, which is based on TFIDF topic representation. This technique rates sentences by calculating their salience based on a set of criteria. The metrics cluster-based relative utility (CBRU) and cross-sentence informational subsumption are defined (CSIS). CBRU determines how significant a certain sentence is to the overall topic of the cluster. To do this, TFIDF vector representations of the documents are produced, and words with TFIDF scores less than a certain threshold are deleted.

➢ Latent Semantic Analysis:

Latent semantic analysis (LSA) is an unsupervised approach for obtaining a semantic representation of text from observed words. One improvement was to use the weight of each subject to determine the size of the summary that should cover a topic.

**Proposed Results**:

The specified input text is the source document. Preprocessing: Tokenization is a technique for dividing text into tokens (words or paragraphs or sentences). Stop words are used to minimize the amount of text; in pre-processing, we have a dictionary made up of stop words. It compares the words in a text and then removes the terms that match. As a result, eliminating stop words will improve performance. Word frequency refers to the most frequently occurring words. In a text, a flow chart is a measure of information. It is calculated by dividing the number of times a word appears in the archive by the total number of words in the archive. The length of sentences is used to remove sentences that are excessively long or too short. It is calculated by multiplying the number of words in the sentence by the number of words in the longest sentence. Sentence scoring and ranking: it assigns a score to each sentence and ranks them accordingly. Extraction of a sentence: The primary goal of this is to find the finest in the text. It is the goal of this exercise to rank whole phrases. Main summary: arrange the phrases in the correct sequence and get the final summary.

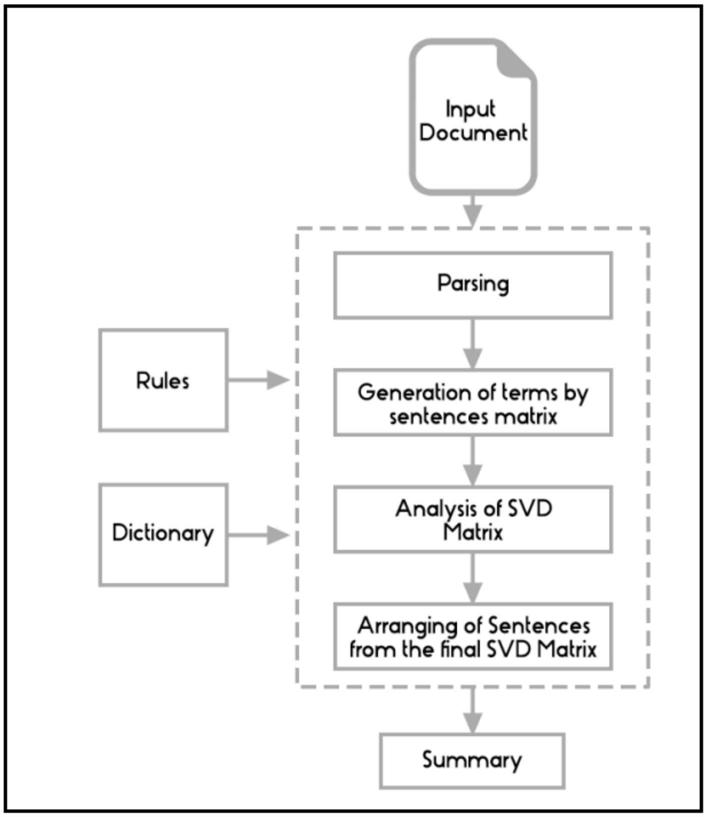
Input Document
Parsing
Rules
Generation of terms by sentences matrix
Dictionary
Analysis of SVD Matrix
Arranging of Sentences from the final SVD Matrix
Summary

*Fig: Process of Summarization*

## CONCLUSION

The research explains how we employ sophisticated strategies for text summarizing on documents utilizing an extractive summarization method termed the Text Rank algorithm. First, we imported the relevant libraries and functions into Python, and then code was written to summarize the text. Following that, a model is presented with small expansions to improve by displaying the outline text. The strategies provided in this research help improve text summarization results using the Genism library in NLP. The general meaning of the paper may therefore be clearly understood.

## Reference's :

1. Hans Peter Luhn, "The automatic creation of literature abstracts", IBM Journal.
2. G. Vijay Kumar and V. Valli Kumari, "Sliding Window Technique to Mine Regular Frequent Patterns in Data Streams using Vertical Format", IEEE International Conference on Computational Intelligenceand Computing Research, 2012.
3. Shohreh Rad Rahimi, Ali Toofan zadeh Mozhdehi and Mohamad Abdolahi, "An Overview on Extractive Text Summarization".“IEEE 4th International Conference on Knowledge Based Engineering and Innovation" (KBEI) Dec. 22nd, 2017, Iran University of Science and Technology – Tehran, Iran.
4. Potharaju, S. P., & Sreedevi, M. (2017). A Novel Clustering Based Candidate Feature Selection Framework Using Correlation Coefficient for Improving Classification Performance. Journal of Engineering Science & Technology Review, 10(6)
5. Potharaju, S. P., & Sreedevi, M. (2017). A Novel Clustering Based Candidate Feature Selection Framework Using Correlation Coefficient for Improving Classification Performance. Journal of Engineering Science & Technology Review, 10(6)
6. Moratanch, N., & Chitrakala, S. (2017, January). A survey on extractive text summarization. In 2017 international conference on computer, communication and signal processing (ICCCSP) (pp. 1-6).
7. Vanetik, N., Litvak, M., Churkin, E., & Last, M. (2020). An unsupervised constrained optimization approach to compressive summarization. Information Sciences, 509, 22-35.
8. Chu, E., & Liu, P. (2019, May). MeanSum: a neural model for unsupervised multi-document abstractive summarization. In International Conference on Machine Learning (pp. 1223-1232)
9. Dohare, S., Gupta, V., & Karnick, H. (2018, July). Unsupervised semantic abstractive summarization. In Proceedings of ACL 2018, Student Research Workshop (pp. 74-83).
10. Gonçalves, Luís. 2020. "Automatic Text Summarization with Machine Learning — An overview." Medium.com
11. Khatri, C., Singh, G., & Parikh, N. (2018). Abstractive and extractive text summarization using document context vector and recurrent neural networks. arXiv preprint arXiv:1807.08000.
12. Lee, G. H., & Lee, K. J. (2017, November). Automatic text summarization using reinforcement learning with embedding features. In Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 2: Short Papers) (pp. 193-197).
13. Schumann, R. (2018). Unsupervised abstractive sentence summarization using length controlled variational autoencoder. arXiv preprint arXiv:1809.05233.
14. Zheng, C., Wang, H. J., Zhang, K., & Fan, L. (2020). A Baseline Analysis for Podcast Abstractive Summarization. arXiv preprint arXiv:2008.10648.
15. Yang, Z., Zhu, C., Gmyr, R., Zeng, M., Huang, X., & Darve, E. (2020). TED: A Pretrained Unsupervised Summarization Model with Theme Modeling and Denoising. arXiv preprint arXiv:2001.00725.
16. R. M. Alguliev, R. M. Aliguliyev, M. S. Hajirahimova, and C. A. Mehdiyev, "Mcmr: Maximum coverage and minimum redundant text summarization model," Expert Systems with Applications, vol. 38, no. 12, pp. 14 514–14 522, 2011.

**17.** S. Pouriyeh, S. Vahid, G. Sannino, G. De Pietro, H. Arabnia, and J. Gutierrez, “A comprehensive investigation and comparison of machine learning techniques in the domain of heart disease,” in Computers and Communications (ISCC), 2017 IEEE Symposium on. IEEE, 2017, pp. 204–207.

**18.** E. D. Trippe, J. B. Aguilar, Y. H. Yan, M. V. Nural, J. A. Brady, M. Assefi, S. Safaei, M. Allahyari, S. Pouriyeh, M. R. Galinski, J. C. Kissinger, and J. B. Gutierrez, “A Vision for Health Informatics: Introducing the SKED Framework.An Extensible Architecture for Scientific Knowledge Extraction from Data,” ArXiv e-prints, 2017.

**19.** M. Allahyari, S. Pouriyeh, K. Kochut, and H. R. Arabnia, “A knowledge-based topic modeling approach for automatic topic labeling,” INTERNATIONAL JOURNAL OF ADVANCED COMPUTER SCIENCE AND APPLICATIONS, vol. 8, no. 9, pp. 335–349, 2017.

**20.** S. Pouriyeh, M. Allahyari, K. Kochut, G. Cheng, and H. R. Arabnia, “Es-lda: Entity summarization using knowledge-based topic modeling,” in International Joint Conference on Natural Language Processing (IJCNLP), 2017.